ESD-TR-72-402

MTR-2462

PROBLEM STATEMENT LANGUAGE AND ANALYZER CONCEPTS AND RECOMMENDATIONS

R. J. Fortin

MARCH 1973

Prepared for

DEPUTY FOR COMMAND AND MANAGEMENT SYSTEMS
ELECTRONIC SYSTEMS DIVISION
AIR FORCE SYSTEMS COMMAND
UNITED STATES AIR FORCE
L. G. Hanscom Field, Bedford, Massachusetts

Project 572F

Prepared by

THE MITRE CORPORATION
Bedford, Massachusetts

Contract No. 19628-73-C-0001

ESD-TR-72-402

MTR-2462

PROBLEM STATEMENT LANGUAGE AND ANALYZER CONCEPTS AND RECOMMENDATIONS

R.J. Fortin

MARCH 1973

Prepared for

DEPUTY FOR COMMAND AND MANAGEMENT SYSTEMS
ELECTRONIC SYSTEMS DIVISION
AIR FORCE SYSTEMS COMMAND
UNITED STATES AIR FORCE
L. G. Hanscom Field, Bedford, Massachusetts

Project 572F

Prepared by

THE MITRE CORPORATION
Bedford, Massachusetts

Contract No. 19628-73-C-0001

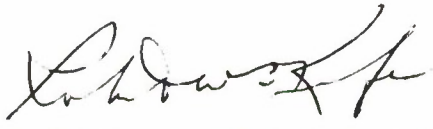
FOREWORD

This work was conducted in support of Project 572F by The MITRE Corporation, Bedford, MA., under Contract F19628-73-C-0001. The need for computer aids to assist in the statement and analysis of Command and Control Information Systems requirements has been clearly established. Indeed, the AFSC Development Plan Study, "Information Processing/Data Automation Implications of Air Force Command and Control Requirements in the 1980s (CCIP-85)", has cited this as one of the five most critical areas in need of further research and development.

A necessary first step toward achieving an adequate requirements analysis capability is the effort initiated by the Directorate of Information Systems Technology, Electronic Systems Division, (ESD/MCI) and MITRE in FY 1972 to survey the current techniques for requirements analysis and to take steps to apply them to Air Force needs. That effort resulted in the preliminary decision to concentrate on the PSL/PSA approach which is the subject of this report. The next step in this effort is the test application of PSL/PSA to an Air Force Information Processing System design problem, to be conducted cooperatively by ESD/MCI and MITRE. This report outlines a procedure for carrying out this effort which insures that the results will provide a sound basis for an evaluation of PSL/PSA as a requirements definition and analysis tool. In addition the recommended approach insures that the results will be of direct benefit to the designer of the system.

REVIEW AND APPROVAL

This technical report has been reviewed and is approved.



ROBERT W. O'KEEFE, Lt Col, USAF
Chief, Field Support Division
Deputy for Command and Management Systems

ABSTRACT

The Information System Design and Optimization System (ISDOS), under development at the University of Michigan, offers automated assistance for the design of information processing systems (IPS). The Problem Statement Language (PSL) and Problem Statement Analyzer (PSA) are the components of ISDOS which are used for IPS requirements definition and analysis. This report describes the ISDOS Project in general and the concepts, capabilities, and use of PSL/PSA. The report also describes future PSL/PSA development plans and makes recommendations for a detailed study to determine PSL/PSA's functional value.

TABLE OF CONTENTS

LIST OF ILLUSTRATIONS

SECTION I

INTRODUCTION

GENERAL

The Information System Design and Optimization System (ISDOS) is being developed as an automated aid for the design of information processing systems (IPS). The University of Michigan has developed this system for the past three years under the direction of Dr. Daniel Teichroew, assisted by his colleagues and graduate students in the Department of Industrial and Operations Engineering.

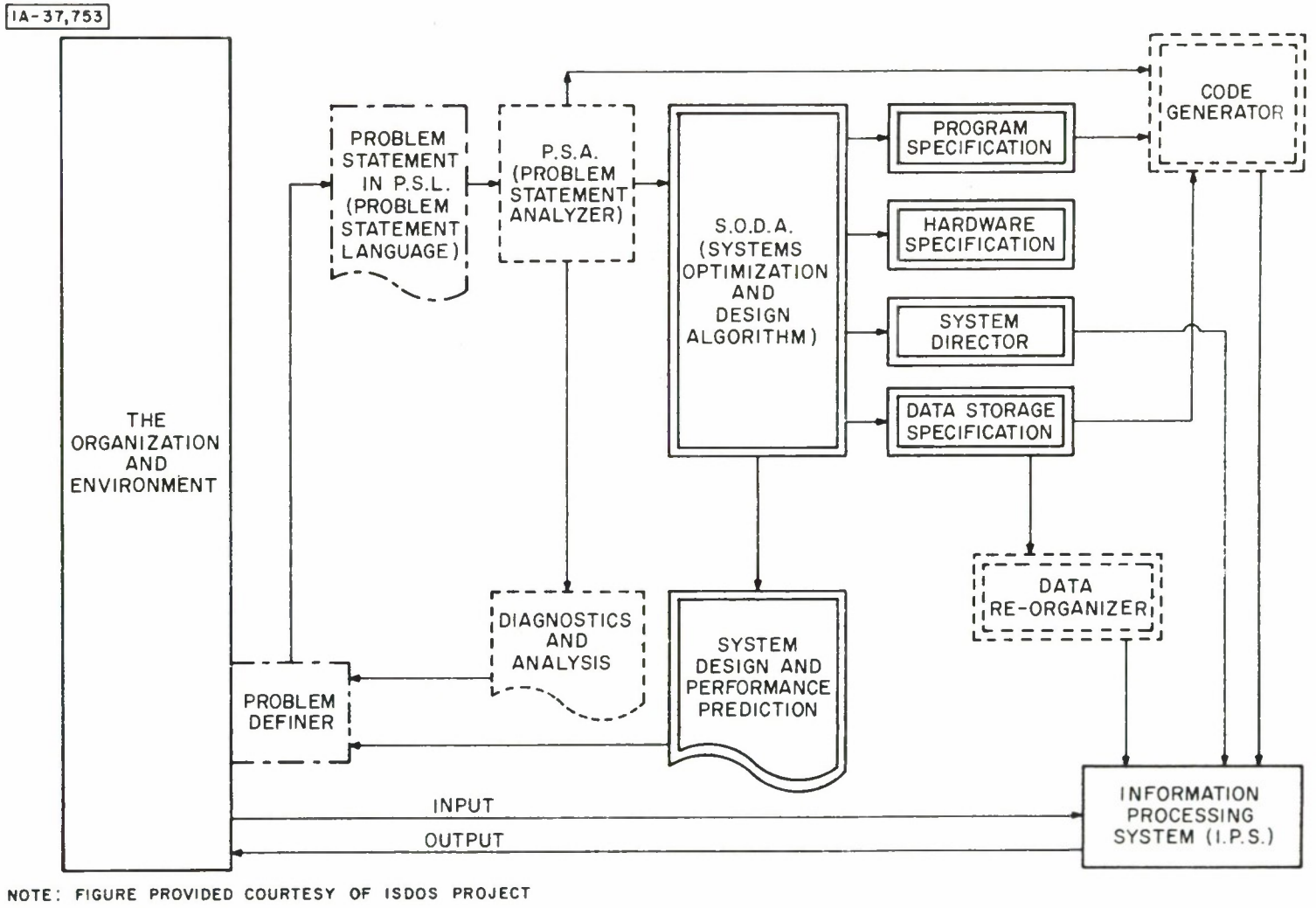
Part of Project 572F is concerned with the investigation of requirements definition and analysis. The ISDOS Problem Statement Language (PSL) and Problem Statement Analyzer (PSA) appear to offer automated assistance for the requirements task.[3] This paper covers the concepts, capabilities and use of PSL and PSA. PSL and PSA should be investigated and evaluated to determine their potential value.

ISDOS STRUCTURE

Figure 1 outlines the ISDOS structure. Both manual and automated system components are combined to produce software and hardware specifications for an information processing system. The Problem Definer uses PSL to state IPS requirements which are then processed by PSA.[4] Diagnostic and analytical reports are used by the Problem Definer to refine the PSL requirements. PSA provides inputs to the Code Generator and the Systems Optimization and Design Algorithm (SODA) which interact with the Problem Definer through feedback loops to finally produce the IPS design.

ISDOS STATUS

The PSA software package has been largely completed and processes most of the PSL statements to produce the planned diagnostic and analytical reports. PSL/PSA may be considered as a stand-alone system in addition to being a component of ISDOS. The SODA software package is in an experimental status. Since SODA requires 1.5 million bytes of core, it is impractical to execute and has no interfaces with PSA. The Code Generator has not been written.



NOTE: FIGURE PROVIDED COURTESY OF ISDOS PROJECT

Figure 1 INFORMATION SYSTEM DESIGN AND OPTIMIZATION SYSTEM (ISDOS)

PSL/PSA USE

PSL/PSA is a major software component of ISDOS used for the requirements definition and analysis task of IPS design. Requirements are stated in PSL and analyzed by PSA. The entire "problem" is segmented into an arbitrary number of subproblems organized in an arbitrary number of levels. Within each subproblem, blocks of PSL statements, termed Problem Statement Units (PSU), are used to express the individual IPS requirements.

Although PSL and PSA were developed initially for experimental purposes, they may be applied to practical IPS applications. With certain modifications (see Appendix I), PSL and PSA could become even more practical for defining and analyzing IPS requirements.

SECTION II

PSL/PSA BACKGROUND

GENERAL

IPS requirements definition and analysis has suffered from a lack of formalized methodology. PSL provides the opportunity to state requirements in a formal, English-related language for a precise and unambiguous meaning. PSA will process the PSL-stated requirements for a lexical, syntactical, and content analysis to produce eight output reports. These reports order, summarize, and interrelate the different requirements.

PSL/PSA DEVELOPMENT

There exist two systems similar in purpose to PSL/PSA. Both of these systems have been studied for the development of PSL/PSA.

The National Cash Register (NCR) Accurately Defined Systems (ADS) is a totally manual system. Requirements are stated in particular formats, but no processing of these statements takes place.

IBM's Time Automated Grid System (TAG) performs a rather extensive analysis of input requirements and produces a number of apparently helpful reports for system design purposes. However, there exists no "language" as such for stating the IPS requirements. In comparison to PSL, rather complicated input tables are used to enter the data.

The developers of PSL/PSA have attempted to include the advantages of ADS and TAG into their system.(3)

VERSIONS OF PSA

Whereas the PSL syntax is rather complete and well defined PSA is lacking in its ability to accept and process a number of PSL statements.(2)

There are two versions of PSA. Version 1 (V1) is a subset of the newer Version 2 (V2) in that V2 recognizes more PSL syntax and produces more reports than V1. V2 is currently implemented on the University of Michigan's IBM 360/67 Dual Processor operating under the Michigan Time-Sharing System (MTS).

PSA V1 has been modified and the necessary Job Control Language (JCL) has been written for implementation under IBM Operating System (OS) with Multiprogramming with a Variable Number of Tasks (MVT). This version is being implemented on The MITRE Corporation's IBM 370/155 computer.

ISDOS personnel at the University of Michigan have tentative plans to modify V2 for OS/MVT operation during September. The additional capabilities of V2 in comparison to V1 are highly desirable, and V2 will be implemented on the MITRE computer when V2 modification has been completed.

DOCUMENTATION

A variety of University of Michigan working papers have been written concerning ISDOS concepts and methodology. However, there is no complete explicit documentation for the functional use of PSL/PSA. A "User's Manual" and "Language Primer" are available for V2, but there is no documentation specifically for V1.

SECTION III

PSL/PSA CAPABILITIES

REQUIREMENTS DEFINITION

Using PSL a Problem Definer states the IPS requirements nonprocedurally, thereby segregating existing or suggested IPS procedures.[5] The PSL designers believed that procedures should be a function of the total IPS requirements and hardware configuration; the IPS output should not be prematurely influenced by either implicit or explicit procedures. These nonprocedural requirements are stated in terms of inputs, outputs, historical tables, policies, or functions. After the IPS has been defined in PSL, the PSA analysis is used by the system analyst (assisted by other planned ISDOS software packages) to formulate the IPS design.

LANGUAGE STRUCTURE

The entire IPS is termed a Problem.[1] The Problem is composed of any number of logical sections called Subproblems. Subproblems may be contained within other Subproblems to any depth; any number of Subproblems may occur at each level.

Problem Statement Units (PSU) which are usually groups of either input or output information, are defined within each subproblem. Each PSU contains up to four sections.

The Identification section contains information pertaining to the PSU's author, date, source, destination, and any other descriptive text for traceability purposes. The data in this section is for information purposes only and is not currently processed by PSA.

The Requirements section defines a Principal Data Set (PDS), which is the name of the input, output, or other function for the PSU. Time and volume information for the PDS is listed, in addition to period and occurrence data. For example, a PSU might occur randomly (occurrence) ten times (volume) a day (period) between 8:00 a.m. and 5:00 p.m. (time).

In the Define section, the PDS data element names and collections of data elements, called groups, are listed. The element formats may be included using "pictures," similar to those used in COBOL.

The Compute section is optional and is used to describe computations in which data elements are derived from other data elements. Although PSL provides for arithmetic and Boolean expressions, only the

Function statement is implemented in PSA. This statement identifies those derived data elements which are functions of other elements.

REQUIREMENTS ANALYSIS

PSA processes the IPS requirements written in PSL to produce a combination of syntax diagnostics and analytical reports. Error messages note those statements PSA could not recognize or interpret, and these statements are omitted from any further processing.

A total of eight reports are produced by PSA/V2, the first five of which are implemented in PSA/V1.

The Problem Statement Structure report names the Problem, Subproblems, and PSUs. Line reference numbers are given for all statements and consecutive numbers are assigned to the PSUs.

The PSL Source Listing and Summary Report contains all of the input source statements with line numbering. Diagnostic messages are interspersed with the source statements wherever errors are found. A Summary lists the number of source statements, the number of lines containing errors, and the number of errors found.

The Synonym Table lists all of the synonyms defined in the PSL problem. The type of synonym, global or local, and a PSU reference are given.

The Problem Functions Table lists a source listing line reference number for each function variable name and the associated operand names. The report will list the function name, but currently PSA/V2 will not accept specific arithmetic or Boolean functions or expressions.

The Problem Statement Directory lists all of the user-defined names with their respective line reference number. Each name is identified by type and by its usage in the various PSUs. Error messages are printed for data use inconsistencies; e.g., a data element is output, but it was never input, calculated, and is not a PDS.

The Network Analysis lists alphabetically all PDSs, groups, and functions. All data sets contained within these are then listed. with cross references for groups and functions to indicate their respective elements.

The Unspecified System Parameter Report lists the PSU volume parameters' names, types, and PSU and line reference numbers. Since PSA/V2 currently will not accept values for parameters, all parameters used in the problem definition will necessarily be unspecified and listed on this report.

The Timing Analysis Report indicates the sequence of PSU processing. The cycle and earliest and latest time phrases are listed for each PSU. Warnings are printed for PSUs where a time phrase check failed, such as an output PSU occurring before all required inputs are available.

SECTION IV

FUTURE PSL/PSA DEVELOPMENT PLANS

SCOPE

The University of Michigan ISDOS Project personnel plan to continue development of PSL/PSA; Appendix I contains a listing of the proposed modifications and extensions. Since PSL is quite well formulated, most of the effort will be given to developing PSA.

OBJECTIVE MODIFIED

PSL has been designed for functional applications, but the development of PSA has been more experimental, so far. Experimentation and testing have shown that PSL/PSA satisfy their design concepts in a limited environment. An objective now exists to extend this environment to that of practical applications and functional use so that the potential of PSL and PSA may be realized and used.

PSL ABBREVIATIONS

Currently there are no abbreviations for PSL reserved words and the resulting long source statements are tedious to write for people familiar with the language. There are plans to establish valid abbreviations for at least the most frequently used and lengthy PSL reserved words. The use of these abbreviations would be at the discretion of the Problem Definer. In the PSL Source Statement Listing and Structure reports, the abbreviations would be replaced by the complete reserved words.

PSL SYNTAX IMPLEMENTATION

PSA/V2 does not accept the complete set of PSL statements as described in the PSL Language Primer. There are plans to produce an updated version of PSA to accept more of the PSL syntax.

PSA REPORTS

The Format Specification Report, as described in the "PSL Language Primer," will be added to PSA. This capability will allow the problem definer to specify PSU formats. PSA will then produce sample PSUs in the formats described with dummy data fields.

Other analytical reports are also being considered for implementation. However, the particular reports remain unspecified at this time.

SECTION V

CONCLUSIONS

These conclusions are based on a paper study of ISDOS and particularly PSL/PSA documents. The capabilities expressed in the publications do not necessarily correspond with the actual PSA software capabilities. Therefore, these conclusions may be modified as the use of PSL/PSA increases.

PSL should be of definite value for defining IPS requirements for either modifying existing systems or designing new systems. The formalized, English-related language is easy to learn and allows the Problem Definer to state IPS requirements precisely and consistently.

PSL is unique; other systems which aid IPS design, optimization, and evaluation do not have a nonprocedural requirements definition language.

Since PSL is English-like in structure, it is easy to understand and may be learned in one or two weeks. Because of the language's simplicity, it appears desirable that the individuals familiar with the IPS to be designed should learn PSL and use the language to define the requirements. This is in contrast to the more conventional procedure where the system analysts would first become familiar with the future IPS design requirements and then define them for analysis purposes.

The value of the PSA output reports for analytical purposes is not clear. The reports certainly reflect and list the input PSL-defined requirements in a number of ways with appropriate cross references, but there may be insufficient analysis for IPS design (e.g., file design, terminal configuration design, security consideration). For example, in the design of many systems, there would be a need to list separately all PDSs having a given security classification.

PSL/PSA are part of a developing system. Necessarily much of their implementation has been experimental, rather than practical in nature, and PSL/PSA has not as yet been used in an actual IPS design project. When functional language and analyzer modifications have been implemented PSL/PSA should become a practical and valuable design tool.

SECTION VI

RECOMMENDATIONS

The initial investigation of ISDOS and particularly PSL/PSA has been completed. A detailed study to determine PSL/PSA's functional value will be undertaken. To accomplish this analysis, the following recommendations are made.

IBM 370/155 PSA IMPLEMENTATION

Since PSA was not designed explicitly to operate under OS/MVT, which is used on The MITRE Corporation's IBM 370/155, JCL modification is required for implementation on the MITRE computer. PSA/V1 has been modified for OS, but because PSA/V2 has additional capabilities it should be used during PSL/PSA evaluation. ISDOS project personnel have plans to modify PSA/V2 for OS/MVT operation during September 1972. Upon modification completion PSA/V2 should be installed in MITRE's 370/155. This version will be implemented at the MITRE Computer Facility.

SAMPLE PSL/PSA EXECUTION

A sample problem has been written in PSL to be processed by PSA. This exercise has been undertaken to gain familiarity and experience with the PSL language, and with PSA diagnostics, output reports, execution times, and costs. Differences between PSA/V1 and V2 processing will also be observed.

PSL/PSA TEST APPLICATION

The utility of the ISDOS Problem Statement Language (PSL) and the Problem Statement Analyzer (PSA) should be measured through the application of PSL/PSA for requirements definition and analysis of an Air Force Information Processing System (IPS) design problem. The size of the problem should be large enough to provide sufficient data for evaluation. Also, it is felt that a more meaningful evaluation will be achieved if Air Force personnel participate directly in the sample problem.

MITRE, in cooperation with ESD, should choose a sample problem. The application of PSL/PSA to the sample problem will be specified by clearly defining the work to be performed. The necessary evaluation procedures and forms will be developed before execution of the sample problem begins. The evaluation will attempt to determine qualitative attributes such as PSL's ease of use, completeness, diagnostics, and output reports.

"Ease of use" must be measured in relative terms by comparison to other requirements definition methods. PSL ease of use will depend upon this language's inherent structure and syntax, the problem definer's familiarity of the language, and the particular problem application.

The "completeness" of PSL depends on this language's capability of providing adequate provisions for defining particular IPS requirements. It is hoped that extraneous, procedural-related information will be excluded, while still including the essential information for IPS design.

The diagnostics should be evaluated for completeness. All PSL input inconsistencies will be noted with ample explanation of the error. Also, warnings will be listed for possible errors such as missing "END" statements.

Finally, the PSA analytical output reports should be evaluated regarding their actual use and value. The reports should sufficiently assist both the Problem Definer in analyzing and modifying requirements, and the analyst in designing the system. Comparisons with reports and analytical results of other IPS design methods may be made.

Air Force personnel should participate directly by stating the IPS requirements in PSL and analyzing the PSA reports. Undoubtedly, several executions will be required for debugging purposes and requirements modification. A minimum of three months will be required for PSL/PSA training and for problem definition, requirements statement, program execution, and analysis purposes. During the first month, Air Force personnel would receive formal PSL training and a general familiarization with PSA operation and the output reports. The sample problem coding and debugging would be accomplished during the second month, and report analysis and requirements modification during the third. Additional time will be required for the evaluation of the sample problem results and final report preparation. It is estimated that 3 to 4 Air Force personnel will be required.

The overall effort of PSL/PSA training, programming, execution, and evaluation will be influenced by the personnel chosen as problem definers. PSL has been designed for usage by those people familiar with the particular IPS requirements, and not necessarily programmers or analysts. The "English-like" language syntax may be easily learned and used by "functional" people to provide a formal, comprehensive, and understandable listing of requirements to the analyst.

The results of this application of PSL/PSA are expected to provide a measure of the utility of such a tool in defining requirements for various Air Force automation system problems. In addition, the results will indicate particular modifications to improve PSL/PSA as a requirements definition and analysis tool. Finally, the results of the actual analysis of the problem's requirements will be of direct benefit to the designers of the system.

APPENDIX I

PSL/PSA DEVELOPMENT RECOMMENDATIONS

Mutual agreement was reached between The MITRE Corporation, other active PSL/PSA users, and ISDOS Project personnel that the University of Michigan would support the following modifications and extensions to PSL/PSA. An ISDOS Newsletter will be published describing these items in detail with expected completion dates.

HIGH PRIORITY

1. Modify the current version of PSA for IBM OS/MVT execution during September. This will permit the implementation of the most recent version of PSA (Version 2) on MITRE's 370/155.

2. Extend PSA to permit the production of "mockup" or sample reports.

3. Implement the "COPY" statement for PSL elements and groups. This will reduce the amount of repetition currently required in the definition of a PSL Problem.

4. Allow abbreviations for PSL reserved words.

5. Implement the "COMPUTE" statement.

MEDIUM PRIORITY

1. Reduce the variable name length from 60 to 30 characters, with an option for other lengths as specified at system generation time.

2. Print more diagnostics, particularly warnings, in the Directory Listings; e.g., notification of different "picture" specifications for the same element.

3. Remove the "UPDATE" statement and use the "HISTORY" function for specifying master files.

4. Implement "SYNONYM" statements to work as intended (U. of M. Working Paper #35), including all names being printed in reports and the synonym in the source listing.

5. Produce cross references between PSUs for the input, history, and output functions.

LOW PRIORITY

1. Add an "IDENTIFICATION" section to the "SUBPROBLEM" statement.

2. Delete the "PDS" statement since it is a repetition, in essence, of the "PSU."

3. Produce a second "source" listing in a formatted form with PSL reserved word abbreviations expanded to full words.

4. Expand the implementation of the "IF" statement.

5. Replace "VOLUME OCCURRENCE" with "DISTRIBUTION."

6. Make the use of "END" statements optional (with warnings printed).

7. Produce decision tables from "IF" statements.

REFERENCES

ISDOS WORKING PAPERS

1. D. Devaux, User's Guide to PSL/PSA, WP-52, November 1971, 67 pps.

2. R. F. Koch, M. J. Krohn, P. W. McGrew, E. H. Sibley, PSL Version 2, Release 1: A PSL Language Primer, WP-33, August 1970, 180 pps.

3. D. Teichroew, Computer-Aided Documentation of User Requirements, WP-39, June 1971, 16 pps.

4. D. Teichroew, Problem Statement Analysis: Requirements for the Problem Statement Analyzer, WP-43, April 1971.

5. D. Teichroew, A Survey of Languages for Stating Requirements for Computer Based Information Systems, WP-42, April 1971, 39 pps.

Security Classification

DOCUMENT CONTROL DATA - R & D

(Security classification of title, body of abstract and indexing annotation must be entered when the overall report is classified)

1. ORIGINATING ACTIVITY *(Corporate author)* The MITRE Corporation P.O. Box 208 Bedford, Massachusetts 01730	2a. REPORT SECURITY CLASSIFICATION UNCLASSIFIED 2b. GROUP

3. REPORT TITLE

PROBLEM STATEMENT LANGUAGE AND ANALYZER CONCEPTS AND RECOMMENDATIONS

4. DESCRIPTIVE NOTES *(Type of report and inclusive dates)*

5. AUTHOR(S) *(First name, middle initial, last name)*

R. J. Fortin

6. REPORT DATE MARCH 1973	7a. TOTAL NO. OF PAGES 21	7b. NO. OF REFS 5
8a. CONTRACT OR GRANT NO. F19628-73-C-0001 b. PROJECT NO. 572F c. d.	9a. ORIGINATOR'S REPORT NUMBER(S) ESD-TR-72-402 9b. OTHER REPORT NO(S) *(Any other numbers that may be assigned this report)* MTR-2462	

10. DISTRIBUTION STATEMENT

Approved for public release; distribution unlimited.

11. SUPPLEMENTARY NOTES	12. SPONSORING MILITARY ACTIVITY Deputy for Command and Management Systems Electronic Systems Division, AFSC L. G. Hanscom Field, Bedford, Mass. 01730

13. ABSTRACT

The Information System Design and Optimization System (ISDOS), under development at the University of Michigan, offers automated assistance for the design of information processing systems (IPS). The Problem Statement Language (PSL) and Problem Statement Analyzer (PSA) are the components of ISDOS which are used for IPS requirements definition and analysis. This report describes the ISDOS Project in general and the concepts, capabilities, and use of PSL/PSA. The report also describes future PSL/PSA development plans and makes recommendations for a detailed study to determine PSL/PSA's functional value.

DD FORM 1 NOV 65 1473

Security Classification

14 KEY WORDS	LINK A		LINK B		LINK C	
	ROLE	WT	ROLE	WT	ROLE	WT
ISDOS						
PSA						
PSL						
REQUIREMENTS ANALYSIS						
REQUIREMENTS DEFINITION						